AMENDMENT TO THE CLAIMS

The listing of the claims will replace all prior listings of claims in the application.

Listing of Claims

1. (Currently Amended) An execution control apparatus of a data driven information processor~~, wherein a handled instruction includes N + 2 (N is an arbitrary integer of at least 1) inputs, and one of the inputs is a constant when an instruction has N + 2 inputs~~, said execution control apparatus comprising:

a handled instruction that includes N + 2 (N is an arbitrary integer of at least 1) inputs, in which of the inputs is a constant when an instruction has N + 2 inputs;

an instruction decoder that decodes an instruction in an input packet and outputs a number of inputs required for said instruction;

a waiting storage region including

a waiting data storage region that can store N waiting data in each waiting data address, and

a data valid flag storage region that stores a data valid flag for each waiting data address, said data valid flag indicating whether the N waiting data stored in said each waiting data address is respectively valid or invalid;

a constant storage device including

a region that stores a constant, and

a constant valid flag storage region that stores a constant valid flag representing whether a constant stored in ~~each~~ a plurality of constant ~~address is~~ addresses are valid or invalid;

a constant readout unit that accesses said constant storage region according to ~~the~~ each constant address information included in the input packet to read out a constant and a constant valid flag from a relevant constant address in said constant storage region;

a waiting operation determination unit that determines a hash address by a hash calculation from contents of the input packet, selects one predetermined way out of a plurality of predetermined ways of processing waiting data, outputs a select signal for the predetermined way of processing waiting data depending upon a combination of a data valid flag for said determined hash address, a constant valid flag read out by said constant readout unit, and the number of instruction inputs output from said instruction decoder for said waiting data storage region, and updates the data valid flag for said hash address based on the select predetermined way of processing waiting data; and

a waiting region access unit being responsive to said select signal to implement a waiting process corresponding to said select signal.

2. (Original) The apparatus according to claim 1, wherein said constant storage region includes a first constant storage region that stores constant data of a first type, and a second constant storage region that stores constant data of a second type,

wherein said constant readout unit identifies whether the readout constant is of said first type or said second type according to the address.

3. (Original) The apparatus according to claim 2, wherein

said constant data of the first type is a scalar constant, and

said constant data of the second type is a vector constant.

4. (Previously Presented) The apparatus according to claim 3, wherein

each input packet can store plurality of data, and

said waiting operation determination unit can store a plurality of data for each packet.

5. (Original) The apparatus according to claim 2, wherein

said constant data of the first type is a scalar constant of a first length, and

said constant data of the second type is a scalar constant of a second length different from said first length.

6. (Original) The apparatus according to claim 1, wherein N of said data valid flags are prepared for one address.

7. (Original) The apparatus according to claim 6, wherein each data valid flag is prepared of one bit for one waiting data of one address, and

said data valid flag storage region includes N flip-flop circuits for each address, each flip-flop circuit storing a data valid flag of one bit.

8. (Original) The apparatus according to claim 1, wherein said data valid flag storage region includes an erasable storage circuit that clears the region in response to a reset signal.

9. (Original) The apparatus according to claim 8, wherein each of data valid flag is prepared of one bit for one waiting data of one address, and

said erasable storage circuit includes a D flip-flop circuit for each address, each D flip-flop circuit storing a data valid flag of one bit.

10. (Original) The apparatus according to claim 1, further comprising an input number detection unit that, when the number of inputs is N+2, updates the constant valid flag to a value representing "invalid", updates the number of inputs to N + 1, and outputs the updated constant valid flag and the updated number of inputs to the waiting processing unit.

11. (Original) The apparatus according to claim 1, wherein N = 2.

12. (Original) The apparatus according to claim 1, wherein N = 1.

13. (Currently Amended) An execution control method of a data driven information processor~~, wherein a handled instruction includes N + 2 (N is an arbitrary integer of at least 1) inputs, and one of the inputs is a constant when an instruction has N + 2 inputs~~, said data driven information processor comprising:

a handled instruction that includes N + 2 (N is an arbitrary integer of at least 1) inputs, in which one of the inputs is a constant when an instruction has N + 2 inputs;

an instruction decoder that decodes an instruction in an input packet to output a number of inputs required for said instruction;

a waiting storage region including

a waiting data storage region that can store N waiting data in each waiting data address, and

a data valid flag storage region that stores a data valid flag for each waiting data address, said data valid flag indicating whether the N waiting data stored in said each waiting data address is respectively valid or invalid;

a constant storage device including

a region that stores a constant, and

a constant valid flag storage region that stores a constant valid flag representing whether a constant stored in ~~each~~ a plurality of constant ~~address is~~ addresses are valid or invalid;

a constant readout unit accessing said constant storage region with a node number of the input packet as a constant address to read out a constant and a constant valid flag from a relevant constant address in said constant storage region;

a waiting operation determination unit that determines a hash address by a hash calculation from contents of the input packet, selects one predetermined way out of a plurality of predetermined ways of processing waiting data, outputs a select signal for the predetermined way of processing waiting data corresponding to a combination of a data valid flag for said determined address, a constant valid flag read out by said constant readout unit, and the number of instruction inputs output from said instruction decoder for said waiting storage region, and updating the data valid flag for said hash address based on the selected predetermined way of processing waiting data; and

a waiting region access unit being responsive to said select signal to implement a waiting process corresponding to said select signal,

said method comprising ~~the steps of~~:

decoding an instruction, wherein an instruction in the input packet is decoded by said instruction decoder, and the number of inputs required by the instruction is output;

reading out a constant, wherein said constant storage region is accessed based on each address information included in the input packet, and a constant and a constant valid flag are read out from a relevant constant address in said constant storage region;

determining a waiting process, wherein an hash address is determined by hash calculation from contents in the input packet, one predetermined way out of a plurality of predetermined ways of processing waiting data is selected, a select signal for the predetermined way of processing waiting data is output corresponding to a combination of a data valid flag for said determined hash address, a constant valid flag read out from said constant readout unit, and the number of instructions output from said instruction

decoder for said waiting storage region, and the data valid flag is updated corresponding to said hash address based on the selected predetermined way of processing waiting data; and

executing the waiting process, wherein, in response to said select signal, a waiting process corresponding to said select process is performed.

14. (Original) The method according to claim 13, wherein said constant storage region includes a first constant storage region that stores constant data of a first type, and a second constant storage region that stores constant data of a second type,

wherein said ~~step of~~ reading out a constant includes ~~the steps of~~:

determining as to whether the constant is of said first type or said second type based on the address, and

reading out the constant.

15. (Original) The method according to claim 14, wherein

said constant data of the first type is a scalar constant, and

said constant data of the second type is a vector constant.

16. (Original) The method according to claim 14, wherein

said constant data of the first type is a scalar constant of a first length, and

said constant data of the second type is a scalar constant of a second length differing from said first length

17. (Original) The method according to claim 13, said data valid flag storage region including an erasable storage circuit clearing the region in response to a reset signal,

said method further comprising the step of applying a reset signal to said storage circuit, thereby clearing said data valid flag storage region.

18. (Original) The method according to claim 13, further comprising ~~the step of~~ updating said constant valid flag to a value representing "invalid" and said number of inputs to N + 1 and applying said updated values to said waiting processing unit when said number of inputs is N + 2, after said ~~step of~~ reading out a constant and before said step of determining a waiting process.

19. (Original) The method according to claim 13, wherein N = 2

20. (Original) The method according to claim 13, wherein N = 1.